

**EHONEY**

**Ehoney欺骗防御系统**

---

**None**

*licheng*

*None*

## Table of contents

---

1. 我们的故事	3
2. 介绍	4
2.1 什么是Ehoney?	4
2.2 什么是欺骗防御?	4
2.3 解决的问题	5
2.4 与蜜罐不同?	6
2.5 演示视频	7
3. 快速开始	8
3.1 环境要求	8
3.2 部署安装	9
3.3 快速使用	10
4. 功能说明	15
4.1 蜜罐拓扑	15
4.2 蜜罐列表	16
4.3 协议转发	19
4.4 透明转发	20
4.5 探针列表	21
4.6 攻击日志	22
4.7 系统设置	23
5. 模拟攻击	26
5.1 HTTP	26
5.2 SSH	28
5.3 MYSQL	29
5.4 REDIS	30
5.5 TELNET	31
6. 其他	32
6.1 名字解释	32
6.2 FAQ	33

## 1. 我们的故事

e签宝是一个创业公司，需要考虑安全的投入产出比，我们是依赖阿里云，但是用不起阿里云的安全产品，这里不得不吐槽阿里云，都上云了以后，安全作为基础设施是越卖越贵，根本用不起。结合我以前的工作经验，索性自研，所以就努力说服老板招人自己做，老板被忽悠成功了，就招聘7个人左右，初创公司没名气，给不起钱，跟大厂没有竞争优势，招人是件痛苦的事情，我们光建立这个安全团队就花费了2年时间(不仅仅是钱的问题)，2年是730天。这里为什么强调天，因为可能在e签宝安全团队每天都是在战斗。说了这么多(废话比较多)，还没说到为啥要做开源，由于我们公司是做saas的，我们自己就爱买saas产品，我们会做安全评估，发现我们采购的所有产品，都存在很严重的安全，TOB这个行业本身就比较苦逼，更没有精力投入安全，他们也面临跟我们一样的问题，我们比他们好点的是有个明智的老板，所以团队小伙伴们就想着能不能把我们自研的安全产品开源出去，多么单纯的想法啊，做起来才发现TM都是坑！

开源啥东西，怎么开源，其实都不知道，只是一股热情，所以就想到开源老鸟，吴敏，吴博士，后续简称老吴，老吴在开源路上给了我们很大的帮助，这里必须感谢下老吴的无私奉献，革命友谊长存，一直欠老吴去一次花都，某天晚上全体开源小组人员开到老吴创业的办公室，听老吴讲如何开源，老吴拿出了讲了N次的PPT给我们讲了一遍。说来也奇怪，每次听完都有不一样的收获，归结原因可能是老吴每次忽悠的都不一样吧，老吴挺能说的，说了4个多小时，兄弟们听的津津有味，实在忍不住肚子饿，就拉着一票兄弟去吃饭，当时已经23点多了，基本饭店都关门了，最后找到一家火锅店，大家吃了起来，这时大家都信心满满，充满期待，不过就是没喝酒，不过这个晚上也确实不一样，因为Ehoney在这个地方梦想起飞！

回去就开始弄开源，这中间又出了一档子事情，信通院也在搞开源，也在做先进安全能力评估，我这个人就是既要又要还要，就让项目负责人徐吉去申报名信通院的评审，评审时间只有一个月，我们开源软件屁都没有呢，就要参加比赛，硬着头皮上，徐吉为了能够参加比赛，拉上项目组小伙伴郑有乐，开始疯狂加班干，人多有多大胆，地有多大产的精神发挥了，其实在参加比赛前几天这东西流程还没跑通，所以比赛就要延期参加，这个也是我们后续项目的延期的开始，由于参加比赛的都是商业产品，也让徐吉开了眼界，认识到我们产品的不足和改进。

比赛参加完以后，回来就是对系统的又一次大改，定了一个开源发布的时间，这个时间变成了一个遥不可及的时间，项目一直延期，每次的预演都变成一次重构，团队同学也是越做也没信心，连续几个月每天工作强度十几个小时，就是没有结果，光打鸣不下蛋，作为团队负责人的我也是十分焦虑，但是我一直没有动摇开源的信念和对团队的信心，最后我也是坐不住了，开始两条腿走路，找新同学高峰对架构进行重新开发，老版本则让徐吉继续优化，我也亲自上阵，抓项目，设计，架构，文档，体验，每周预演一次，每次都骂，最后骂的大家都生病了，这个项目徐吉流鼻血，郑有乐全身过敏，阿飞想跑路，团队唯一的前端妹子，也被我们辣手摧花，想想单身是有道理的。作为老大的我，内心很无奈，也很感动，这更加坚定，要把东西做出来，做出价值。

今天是我们第一次提交代码，写这篇文章进行纪念，我相信这是一个伟大的产品的开始，“星星之火，可以燎原”，相信这个团队会给安全开源带来不一样的血液，如Indiana Jones，永远不会放弃找寻圣杯，我也不再放弃给自己热爱的事业工作，无论其他选择是如何如何的安全。当我老去甚至挂掉，我回顾我的一生，“wow，真是一个史诗般的冒险故事啊”而不会说“wow，我一生过得真是平稳无比”。最后也请使用我们产品的各位小伙伴，也请给这年轻的团队更多的包容和鼓励！

最后更新: 2021年6月29日

## 2. 介绍

### 2.1 什么是Ehoney ?

Ehoney是e签宝安全团队首次开源的欺骗防御系统. Ehoney是基于云原生的欺骗防御系统、也是业界唯一对标商业系统的开源产品、欺骗防御系统通过部署高交互高仿真蜜罐及流量代理转发、再结合自研密签及诱饵、将攻击者攻击引导到蜜罐中、达到扰乱攻击者以及延迟攻击的效果、可以有效保证业务的正常运行.

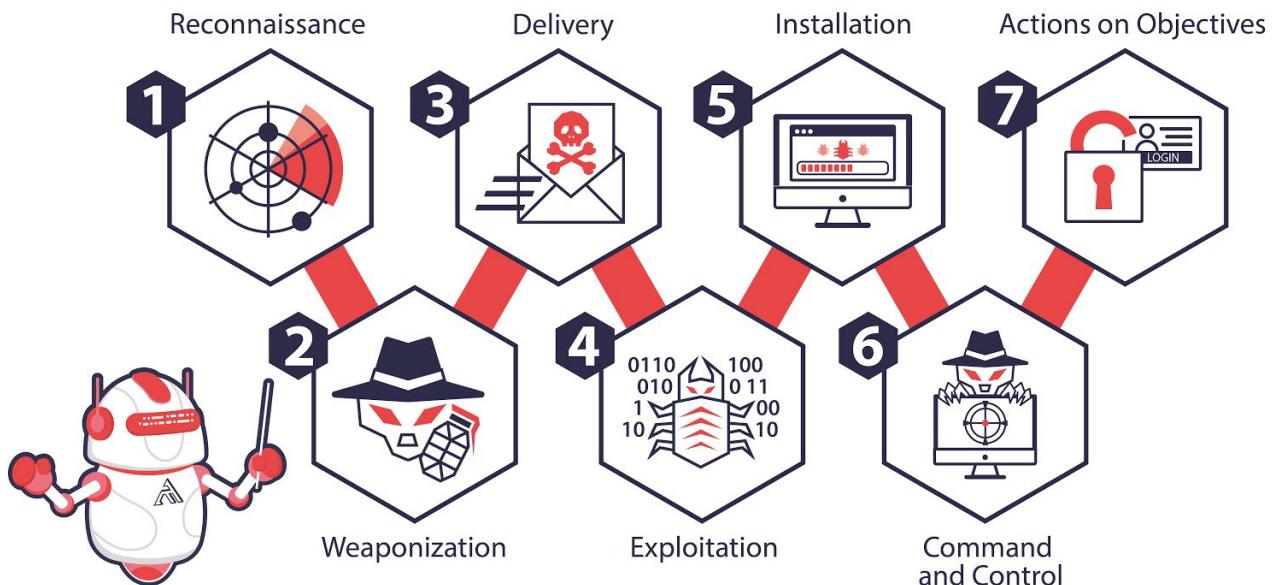
### 2.2 什么是欺骗防御 ?

《孙子兵法》说、兵者、诡道也. 从古至今、欺骗技术作为战场上一种积极防御策略被一直使用. 而网络欺骗技术、就是信息安全战场上防御者的“诡计”. Gartner在2016年安全与风险管理峰会提出的10大信息安全技术中、就包含了欺骗防御技术、Gartner定义欺骗防御技术为“使用骗局或者假动作来阻挠或者推翻攻击者的认知过程、扰乱攻击者的自动化工具、延迟或阻断攻击者的活动、通过使用虚假的响应、有意的混淆、以及假动作、误导等伪造信息达到“欺骗”的目的”. 欺骗技术(Deception Technology )已连续三年被Gartner列为十大安全技术之一、Gartner认为、未来5-10年、欺骗技术将成为主流安全产品对抗未知威胁、0day攻击、高级可持续攻击等安全问题的最佳解决方案。



## 2.3 解决的问题

# THE CYBER KILL CHAIN



Cyber Kill Chain 不仅仅是一种攻击模型。Cyber Kill Chain 的七个阶段为任何组织的安全设计提供了极好的基础。

- **侦察追踪** 攻击者收集有关其目标的信息。这包括间接和被动方法，攻击者通过这些方法从 ARIN（美国互联网号码注册机构）注册、Shodan 或工作列表等公共资源中收集信息。然后攻击者将转向更直接和主动的方法，例如端口扫描。
- **武器构建** 攻击者现在利用侦察中发现的漏洞发起攻击；许多利用来自 metasploit、exploit-db 或社会工程工具包的工具。
- **载荷投递** 一旦攻击者选择了最适合利用您的漏洞的工具，他们就会选择传送方法，无论是网络钓鱼电子邮件、受感染的 USB 还是其他选择的方法。
- **漏洞利用** 武器已交付。攻击者只需要执行攻击，它可以采用 SQL 注入、缓冲区溢出、RCE 以及无数其他形式。
- **安装植入** 攻击者获得更好的访问权限。常见的对抗技术包括在无文件攻击中执行 powershell、安装远程访问工具 (RAT) 和 DLL 劫持。
- **持续控制** 攻击者现在设置对您的系统的持久访问以进行远程操作。根据他们的最终目标，攻击者可以立即采取行动，也可以潜伏在您的系统中数月到数年。即使在重新启动或修补初始漏洞后，访问仍可能持续存在，并且通常被加密和屏蔽以使其看起来像正常流量。有时，它甚至嵌入在正常的合法流量中，例如 Twitter 或电子邮件。
- **目标达成** 攻击者实现他们的入侵目标，例如数据泄露、数据破坏或拒绝服务。

然而，正如网络安全中的一句老话，“防御者需要防御一切，而攻击者只需要利用一个弱点。”，师以长夷以制夷，我们也可以用黑客的手段，反制黑客，打造一套覆盖整个攻击链的欺骗体系

## 2.4 与蜜罐不同？

- **仿真环境** 也就是大家理解的蜜罐，现在市场主流的蜜罐分高中低三类，诉求和解决的问题也不一样，如果要真实模拟环境，高交互蜜罐最合适，实现起来也更复杂，但是市面大部分蜜罐都低交互，稍微有点经验的黑客就能识破，何来欺骗！
- **覆盖率** 蜜罐是被动放在那里，等待黑客自己进来，比如线上服务器1w台，你不可能去部署1w台蜜罐，如果蜜罐仅仅部署几台，犹如杯水车薪，防御效果可想而知，所以这个是欺骗防御必须要解决的问题，就是如何做到高效的请君入瓮
- **攻击溯源** 如果采用高交互蜜罐，黑客入侵进去以后，怎么记录所有黑客的攻击，在蜜罐里装监控，黑客很容易就能发现，而且还能kill该监控，一般黑客都是攻击脚本不落盘，木马程序直接内存运行，没有办法拿到黑客样本，溯源非常困难
- **动态对抗** 蜜罐仅仅只能对攻击进行溯源，分析，不能做到根据黑客的行为，预测黑客的下一步，做到防范于未然，这个不仅仅是分析能力不足，蜜罐的架构也是没法实现动态对抗
- **安全风险** 黑客入侵蜜罐，如果蜜罐没做任何网络隔离，可能就会通过蜜罐做横向渗透测试

### 2.4.1 EHoney特点？

- 支持丰富的蜜罐类型

通用蜜罐：SSH 蜜罐、Http蜜罐、Redis蜜罐、Telnet蜜罐、Mysql蜜罐、RDP 蜜罐 IOT蜜罐：RTSP 蜜罐 工控蜜罐：ModBus 蜜罐

- 基于云原生技术

基于k3s打造saas平台欺骗防御，无限生成蜜罐，真实仿真业务环境

- 业内独一无二密签技术

独创的密签技术，支持20多种密签，如文件、图片，邮件等

- 强大诱饵

支持数十种诱饵，通过探针管理，进行欺骗引流

- 可视化拓扑

可以可视化展示攻击视图，让所有攻击可视化，形成完整的攻击链路

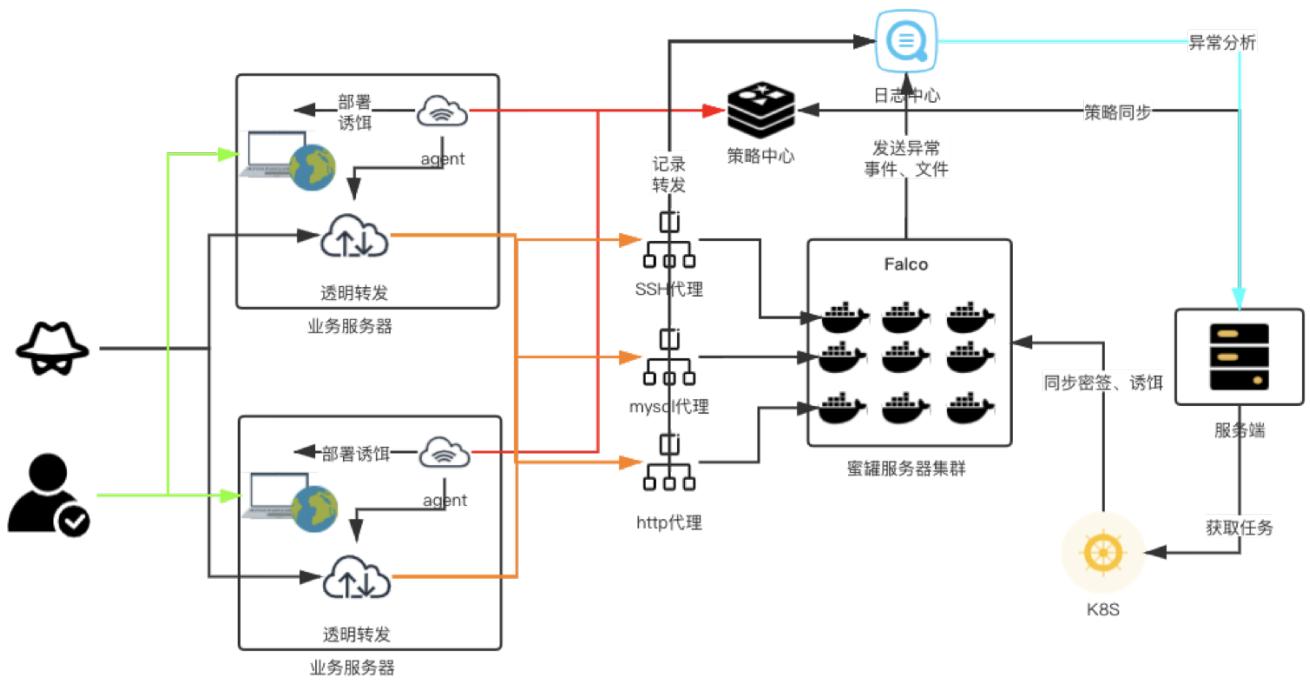
- 动态对抗技术

基于LSTM的预测算法，可以预测黑客下一步攻击手段，动态欺骗，延缓黑客攻击时间，保护真实业务

- 强大的定制化

支持自定义密签、诱饵、蜜罐等，插件化安装部署，满足一切特性需求

## 2.4.2 Ehoney技术架构



## 2.5 演示视频



最后更新: 2021年6月30日

## 3. 快速开始

---

### 3.1 环境要求

#### 3.1.1 环境要求

- **系统要求:** CentOS 7 及以上
  - **最低配置:** 内存4G、磁盘空间10G以上
  - **建议配置:** 内存8G、磁盘空间30G以上
- 

最后更新: 2021年6月30日

## 3.2 部署安装

### 3.2.1 部署安装

```
git clone https://github.com/seccome/Ehoney.git      //clone代码  
cd Ehoney && chmod +x quick-start.sh && ./quick-start.sh //执行一键安装脚本  
输入序号选择服务器IP、安装需等待约10-20分钟（视网络情况而定）
```

安装完成后、访问**http://服务器IP:8080/decept-defense**

默认账号密码如下

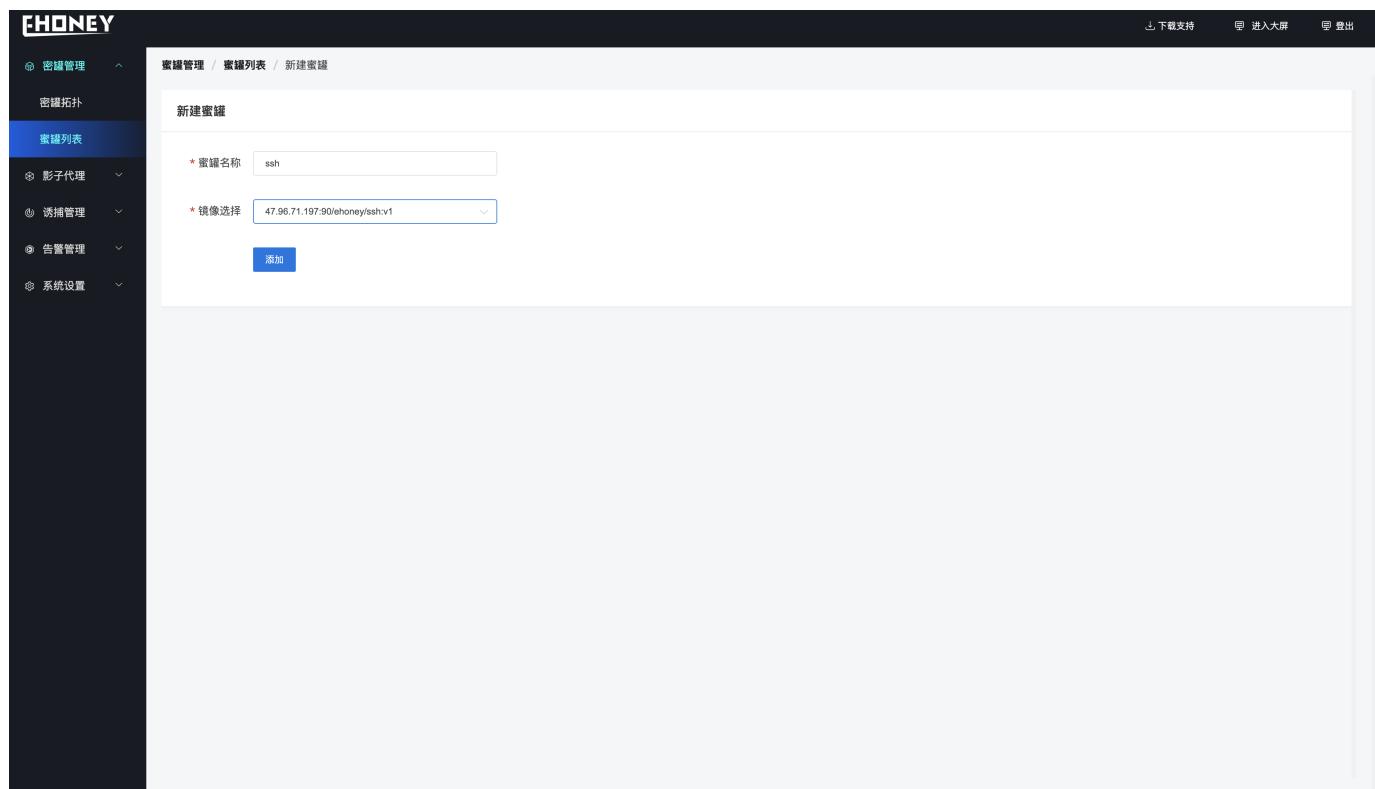
```
账户: admin  
密码: 123456
```

最后更新: 2021年6月30日

## 3.3 快速使用

### 1. 创建蜜罐

蜜罐管理>>>蜜罐列表>>>新建



创建成功后、蜜罐列表显示蜜罐信息、同时可对蜜罐进行下线、存活探测等操作

蜜罐类型	蜜罐名称	蜜罐 IP	蜜网 IP	操作系统	创建时间	下线时间	状态	创建人	操作
telnet	telnet	10.42.0.59	192.168.22.176	Centos	2021-06-28 0:05		在线	admin	密签列表 诱饵列表 下线 网络探测
http	afeitomcat	10.42.0.58	192.168.22.176	Centos	2021-06-28 1:9:39		在线	admin	密签列表 诱饵列表 下线 网络探测
ssh	ssh	10.42.0.57	192.168.22.176	Centos	2021-06-28 1:9:36		在线	admin	密签列表 诱饵列表 下线 网络探测
http	afei2	10.42.0.228	192.168.22.176	Centos	2021-06-28 1:7:59		在线	admin	密签列表 诱饵列表 下线 网络探测
http	afei	10.42.0.227		Centos	2021-06-28 1:7:37	2021-06-28 1:7:58	离线	admin	密签列表 诱饵列表

## 2.部署探针服务器

- 点击右上角的下载支持，将压缩包拷贝到即将部署的服务器上。
- 解压文件tar -zvxf deceipt-agent.tar.gz
- 修改解压目录中的conf目录下的agent.json，修改strategyAddr参数ip为redis地址的ip，默认安装为当前服务器ip 以及strategyPass的值为redis的密码默认为123456。修改sshKeyUploadUrl的ip尾web服务的ip， 默认安装为当前服务器ip。(尽量不要把探针部署在服务端所在的机器。原因是服务端默认已部署了一台探针服务器作为协议代理Agent，agentid 系统唯一)
- 执行 chmod +x deceipt-agent && ./deceipt-agent -mode=EDGE
- 查看启动日志和探针列表是否有此探针服务确认启动是否正常(由于agent端口判断需要，需提前安装lsof)。

```
[root@localhost agent]# chmod +x deceipt-agent && ./deceipt-agent -mode=EDGE
[INFO] [agent/deceipt-agent.go:106] hosteye agent start path: /tmp/agent
fileLogger Init:[{"filename": "log/agent.log", "append": true, "maxlines": 1000000, "maxsize": 10, "daily": true, "level": "INFO", "permit": "0660", "LogLevel": 0}]
2021-06-25 10:34:25 [INFO] [agent/deceipt-agent.go:115] hosteye agent start mode [EDGE]
2021-06-25 10:34:25 [INFO] [util/comm/service.go:491] start mem and cpu monitor, pid: 23576 process: deceipt-agent
2021-06-25 10:34:25 [INFO] [agent/deceipt-agent.go:479] agentid: eeb01ea0-e340-5e67-7285-bc19a39b5120-localhost.localdomain
2021-06-25 10:34:25 [INFO] [agent/deceipt-agent.go:267] start modules
2021-06-25 10:34:25 [DEBUG] [util/comm/service.go:155] process [22684] is [S]
2021-06-25 10:34:25 [DEBUG] [util/comm/service.go:155] process [22685] is [S]
2021-06-25 10:34:25 [DEBUG] [util/comm/service.go:155] process [22684] is [S]
2021-06-25 10:34:25 [DEBUG] [util/comm/service.go:155] process [22685] is [S]
2021-06-25 10:34:25 [DEBUG] [util/comm/service.go:155] process [22685] is [S]
2021-06-25 10:34:25 [DEBUG] [agent/deceipt-agent.go:66] start mem and cpu monitor job
2021-06-25 10:34:25 [INFO] [agent/deceipt-agent.go:497] {"AgentId": "eeb01ea0-e340-5e67-7285-bc19a39b5120-localhost.localdomain", "Status": "running", "Version": "1.0", "": "EDGE"}
2021-06-25 10:34:25 [INFO] [agent/deceipt-agent.go:506] eyJBZ2VudElkIjoiZWViMDFLYTAzZTMC01ZTY3LTcyODUtYmxEOWEzOWI1MTIwLWxvY2FsaG9zdC5sb2NhbGRvbWFpbisIlN0YXRicyYILCJib3N0TmfzS16ImxvY2FsaG9zdC5sb2NhbGRvbWFpbisI1R5cgUi0iJFREdIn0=
2021-06-25 10:34:30 [DEBUG] [util/comm/service.go:519] pid[23576] process [deceipt-agent] overall[0] cpu usage: 0.00 %
2021-06-25 10:34:30 [INFO] [agent/deceipt-agent.go:354] [03583cd75bf401944b018f81b3f6916d-22333 | 22874] cpu usage: 0.00 %, limit threshold [8]
2021-06-25 10:34:30 [INFO] [agent/deceipt-agent.go:354] [81c3b080dad537de7e10e0987a4bf52e-3366 | 8328] cpu usage: 0.00 %, limit threshold [8]
2021-06-25 10:34:30 [INFO] [agent/deceipt-agent.go:354] [86a1b907d54bf7010394bf316e183e67-6381 | 10077] cpu usage: 0.00 %, limit threshold [8]
2021-06-25 10:34:30 [INFO] [agent/deceipt-agent.go:354] [deceipt-bait | 22685] cpu usage: 0.00 %, limit threshold [8]
2021-06-25 10:34:30 [INFO] [agent/deceipt-agent.go:354] [deceipt-proxy | 22684] cpu usage: 0.00 %, limit threshold [8]
2021-06-25 10:34:30 [DEBUG] [util/comm/service.go:519] pid[23576] process [deceipt-agent] overall[0] cpu usage: 0.00 %
2021-06-25 10:34:35 [INFO] [agent/deceipt-agent.go:354] [03583cd75bf401944b018f81b3f6916d-22333 | 22874] cpu usage: 0.00 %, limit threshold [8]
2021-06-25 10:34:35 [INFO] [agent/deceipt-agent.go:354] [81c3b080dad537de7e10e0987a4bf52e-3366 | 8328] cpu usage: 0.00 %, limit threshold [8]
2021-06-25 10:34:35 [INFO] [agent/deceipt-agent.go:354] [86a1b907d54bf7010394bf316e183e67-6381 | 10077] cpu usage: 0.00 %, limit threshold [8]
2021-06-25 10:34:35 [INFO] [agent/deceipt-agent.go:354] [deceipt-bait | 22685] cpu usage: 0.00 %, limit threshold [8]
2021-06-25 10:34:35 [INFO] [agent/deceipt-agent.go:354] [deceipt-proxy | 22684] cpu usage: 0.00 %, limit threshold [8]
```

### 3. 创建协议转发

影子代理>>>协议转发>>>新建

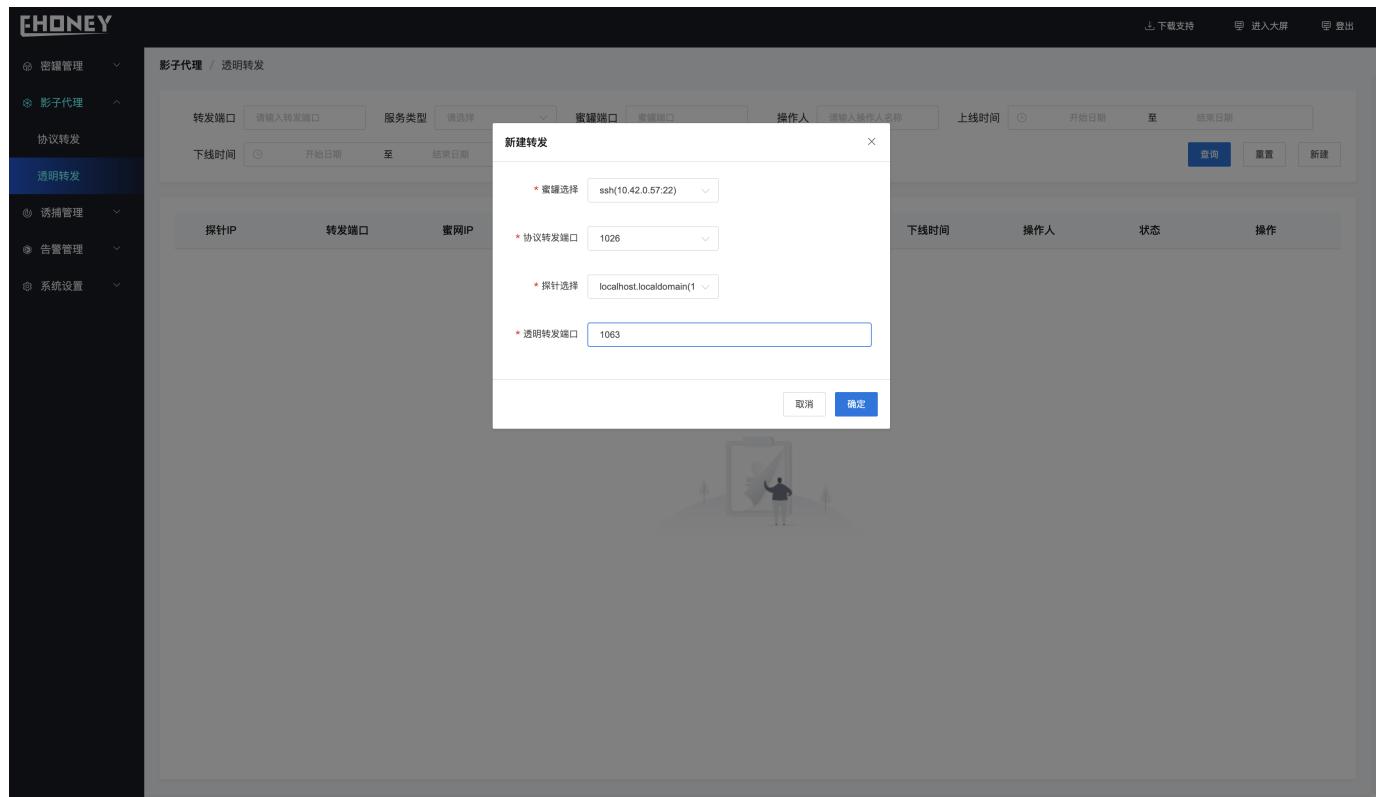
蜜网IP	转发端口	服务类型	蜜罐IP	蜜罐端口	操作人	状态	操作
192.168.22.176	2204	telnet			admin	创建成功	网络探测 下线
192.168.22.176	2203	ssh			admin	创建失败	

创建成功后，显示如下、协议转发表示建立了蜜网端口到蜜罐端口的流量转发、样例创建的协议转发将**1026**端口的**ssh**流量转发到IP地址为**10.42.0.57:22**的蜜罐中

蜜网IP	转发端口	服务类型	蜜罐IP	蜜罐端口	创建时间	下线时间	操作人	状态	操作
192.168.22.176	1026	ssh	10.42.0.57	22	2021-06-29 10:45		admin	创建成功	网络探测 下线
192.168.22.176	2204	telnet	10.42.0.59	23	2021-06-28 20:06		admin	创建成功	网络探测 下线
192.168.22.176	2203	ssh	10.42.0.57	22	2021-06-28 19:46	2021-06-28 19:56	admin	创建失败	

#### 4. 创建透明转发

影子代理>>>透明转发>>>新建



创建成功后、显示如下、透明转发表示建立了探针端口到蜜网端口的流量转发、样例创建的透明转发将**1063**端口的流量转发到蜜网的**1026**端口

The screenshot shows the CHONEY web-based management interface. On the left, there's a dark sidebar with a navigation menu. The '透明转发' (Transparent Forwarding) option is currently selected and highlighted in blue. The main content area is titled '影子代理 / 透明转发'. It contains several search and filter fields at the top, including '转发端口' (Forwarding Port), '服务类型' (Service Type), '蜜罐端口' (Honeypot Port), '操作人' (Operator), '上线时间' (Online Time), and '下线时间' (Offline Time). Below these are buttons for '查询' (Search), '重置' (Reset), and '新建' (New). A table lists the configuration details for a single entry:

探针IP	转发端口	蜜网IP	蜜网端口	服务类型	上线时间	下线时间	操作人	状态	操作
192.168.22.246	1063	192.168.22.176	1026	ssh	2021-06-29 10:53		admin	创建中	<a href="#">网络探测</a>

完成上述几个步骤后、当黑客探测到业务服务器的1063端口并进行攻击时、就会将黑客引诱到ssh服务蜜罐中、同时探针以及蜜罐可以部署密签、诱饵等诱使黑客对蜜罐的攻击、延缓以及防治业务服务被攻击。具体请详读功能说明

最后更新: 2021年7月9日

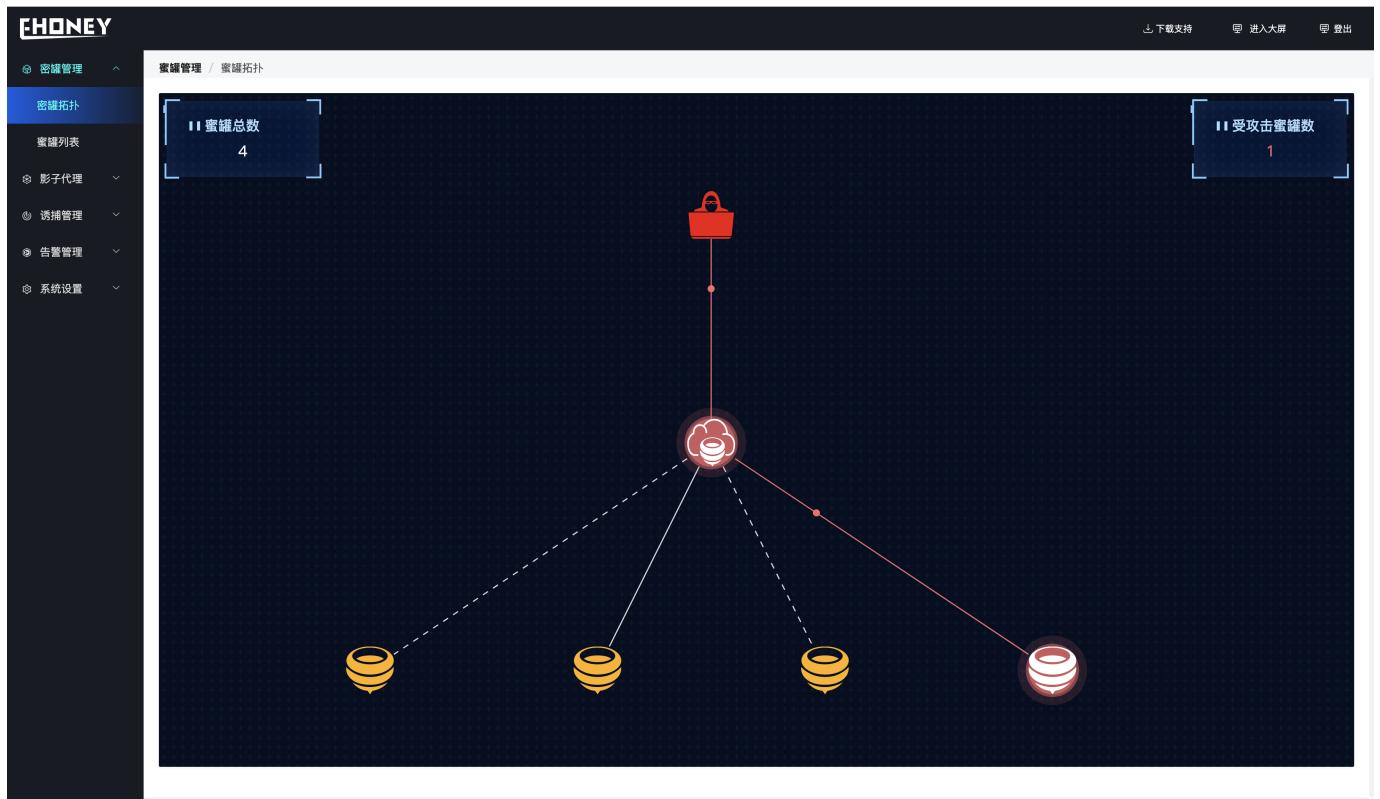
## 4. 功能说明

### 4.1 蜜罐拓扑

#### 4.1.1 蜜罐拓扑

蜜罐管理>>>蜜罐拓扑

显示蜜罐的拓扑图



最后更新: 2021年6月30日

## 4.2 蜜罐列表

### 4.2.1 蜜罐列表

蜜罐管理>>>蜜罐列表

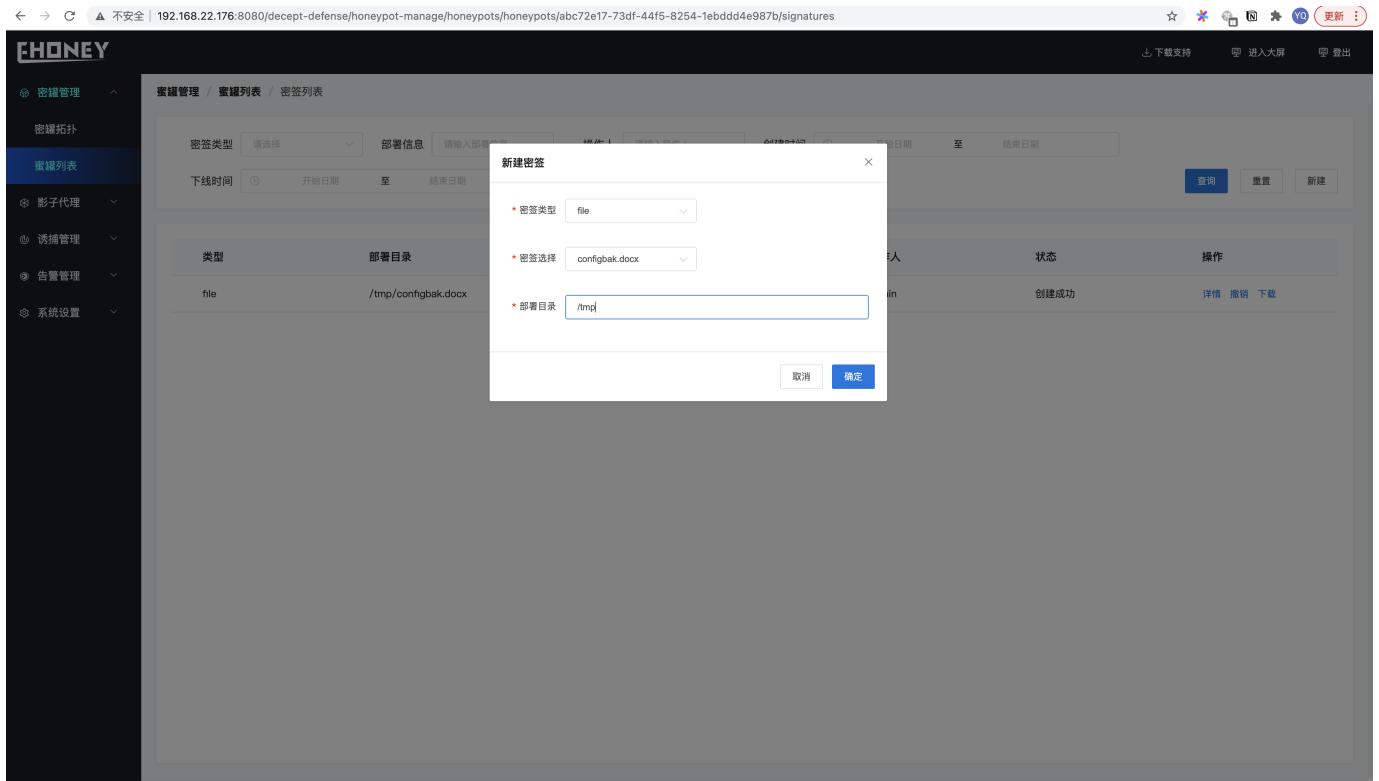
蜜罐类型	蜜罐名称	蜜罐 IP	蜜网 IP	操作系统	创建时间	下线时间	状态	创建人	操作
telnet	telnet	10.42.0.59	192.168.22.176	Centos	2021-06-28 0:05		在线	admin	<a href="#">密签列表</a> <a href="#">诱饵列表</a> <a href="#">离线</a> <a href="#">网络探测</a>
http	afeitomcat	10.42.0.58	192.168.22.176	Centos	2021-06-28 9:39		在线	admin	<a href="#">密签列表</a> <a href="#">诱饵列表</a> <a href="#">离线</a> <a href="#">网络探测</a>
ssh	ssh	10.42.0.57	192.168.22.176	Centos	2021-06-28 9:36		在线	admin	<a href="#">密签列表</a> <a href="#">诱饵列表</a> <a href="#">离线</a> <a href="#">网络探测</a>
http	afei2	10.42.0.228	192.168.22.176	Centos	2021-06-28 7:59		在线	admin	<a href="#">密签列表</a> <a href="#">诱饵列表</a> <a href="#">离线</a> <a href="#">网络探测</a>
http	afei	10.42.0.227		Centos	2021-06-28 7:37	2021-06-28 7:58	离线	admin	<a href="#">密签列表</a> <a href="#">诱饵列表</a>

#### 1. 按照条件查询特定蜜罐信息

2. 新建蜜罐、可以选择不同的镜像创建不同的服务蜜罐（镜像通过harbor管理、后面会介绍如何设置harbor地址）

3. 密签列表、可以部署密签文件到指定的蜜罐中、当密签文件被打开时会显示攻击者的详细信息

新建蜜罐密签时可以选择密签类型（当前支持file类型）、选择密签文件以及部署在蜜罐中的路径



当部署的密签文件被打开时、密签的详情页面会显示打开者的UA、IP等信息

追踪码	追踪文件名称	密签打开ip	ip归属国家	ip归属城市	打开时间	设备指纹
6b415427-1led-43c8-b3c 5-f5f0472d4174	configbak.docx	192.168.22.176			2021-06-29 11:14:10	Microsoft Office Word/16.3 8.614 (Mac OS/10.16; Desktop; zh-CN; NonAppStore; Apple/MacBookPro16,1)
6b415427-1led-43c8-b3c 5-f5f0472d4174	configbak.docx	192.168.22.176			2021-06-29 11:14:11	Microsoft Office Word/16.3 8.614 (Mac OS/10.16; Desktop; zh-CN; NonAppStore; Apple/MacBookPro16,1)
6b415427-1led-43c8-b3c 5-f5f0472d4174	configbak.docx	192.168.22.176			2021-06-29 11:14:11	Microsoft Office Existence Discovery

1. 诱饵列表、新建诱导诱饵、当前支持文件形式诱饵
2. 下线操作可以禁用当前蜜罐
3. 网络探测检测当前蜜罐是否在线

最后更新: 2021年6月30日

## 4.3 协议转发

### 4.3.1 协议转发

协议转发能够将固定的协议流量直接转发到对应的蜜罐中、同时能够对流量进行记录、解析并上报异常信息。

Trap IP	Forward Port	Service Type	Trap IP	Trap Port	Create Time	Offline Time	Operator	Status	Operation
192.168.22.176	1026	ssh	10.42.0.57	22	2021-06-29 10:45		admin	Created successfully	Network probe online
192.168.22.176	2204	telnet	10.42.0.59	23	2021-06-28 20:06		admin	Created successfully	Network probe online
192.168.22.176	2203	ssh	10.42.0.57	22	2021-06-28 19:46	2021-06-28 19:56	admin	Creation failed	Network probe offline

1. 按照指定条件查找协议转发
2. 新建协议转发
3. 探测协议转发服务是否正常
4. 禁用该协议转发服务

最后更新: 2021年6月30日

## 4.4 透明转发

### 4.4.1 透明转发

将攻击者的流量转发到蜜网服务器

**CHONEY**

影子代理 / 透明转发

④ 密罐管理

⑤ 影子代理

协议转发

**透明转发**

⑥ 诱捕管理

⑦ 告警管理

⑧ 系统设置

1

2

探针IP	转发端口	蜜网IP	蜜网端口	服务类型	上线时间	下线时间	操作人	状态	操作
192.168.22.246	1063	192.168.22.176	1026	ssh	2021-06-29 10:53		admin	创建成功	3 4 网络探测 下线

1. 按照指定条件查找透明转发
2. 新建透明转发
3. 探测透明转发服务是否正常
4. 禁用该透明转发服务

最后更新: 2021年6月30日

## 4.5 探针列表

### 4.5.1 探针列表

应用服务器、作为黑客可以感知的第一道入口、探针中可以部署密签、诱饵

The screenshot shows the HONEY web interface. The left sidebar has a dark theme with white text and icons. It includes sections for 密鑑管理, 影子代理, 情報管理, 探針列表 (which is highlighted in blue), 告警管理, and 系統設置. The main content area has a light gray background. At the top, there are search fields for 应用名称 (请输入应用名称), 应用IP (请输入 IP), and 状态 (请选择), along with a dropdown menu and two buttons: '查询' (Query) and '重置' (Reset). Below these is a table with the following data:

Agent ID	应用名称	应用IP	注册时间	心跳时间	状态	操作
eeb01ea0-e340-5e67-7285-bc19a39b5120-localhost.localdomain	localhost.localdomain	192.168.22.246	2021-06-29 10:51:48	2021-06-29 11:33:48	在线	<a href="#">密签列表</a> <a href="#">诱饵列表</a>

最后更新: 2021年6月30日

## 4.6 攻击日志

### 4.6.1 攻击日志

记录黑客的攻击日志，并显示不同协议的详细信息

攻击IP	攻击跳转IP	蜜罐IP	被攻击服务	攻击IP地理位置	攻击时间	操作
192.168.7.232	192.168.22.246	10.42.0.62	redis	局域网-局域网	2021-06-29 14:36	<a href="#">详情</a>
192.168.22.246	192.168.22.176	10.42.0.62	redis	局域网-局域网	2021-06-29 14:36	<a href="#">详情</a>
192.168.7.232	192.168.22.246	10.42.0.59	telnet	局域网-局域网	2021-06-29 14:31	<a href="#">详情</a>
192.168.22.246	192.168.22.176	10.42.0.61	mysql	局域网-局域网	2021-06-29 14:25	<a href="#">详情</a>
192.168.7.232	192.168.22.246	10.42.0.61	mysql	局域网-局域网	2021-06-29 14:25	<a href="#">详情</a>
192.168.7.232	192.168.22.246	10.42.0.61	mysql	局域网-局域网	2021-06-29 14:25	<a href="#">详情</a>
192.168.22.246	192.168.22.176	10.42.0.61	mysql	局域网-局域网	2021-06-29 14:25	<a href="#">详情</a>
192.168.22.246	192.168.22.176	10.42.0.61	mysql	局域网-局域网	2021-06-29 14:24	<a href="#">详情</a>
192.168.7.232	192.168.22.246	10.42.0.61	mysql	局域网-局域网	2021-06-29 14:24	<a href="#">详情</a>
192.168.22.246	192.168.22.176	10.42.0.61	mysql	局域网-局域网	2021-06-29 14:24	<a href="#">详情</a>

共 44 条 < 1 2 3 4 5 > 前往 1 页

最后更新: 2021年6月30日

## 4.7 系统设置

### 4.7.1 系统设置

系统设置可进行镜像源配置、查看镜像列表、进行协议配以及密签跟踪服务器设置

镜像源采用**harbor**为镜像源、版本为**2.0**、指定**harbor的URL**、用户名、密码以及项目名

The screenshot shows the 'Image Source Configuration' page of the HONEY system. On the left is a sidebar with navigation items: 密鑑管理, 影子代理, 誘捕管理, 告警管理, 系统设置 (highlighted), 镜像源配置 (highlighted), 镜像列表, 协议配置, and 密簽配置. The main area has a title '系统设置 / 镜像配置'. It contains four input fields with red asterisks: 'HarborUrl' (empty), '用户名' (empty), '密码' (empty), and '项目名' (empty). Below these is a blue '添加' button. At the bottom is a table with the following columns: 'harbor url', '用户名', '密码', '项目名', and '操作'. The '操作' column contains a small icon. The background features a faint illustration of a person standing next to a large rectangular object.

注意：如果设置自定义harbor 需要修改 /usr/lib/systemd/system/docker.service 文件 设置 ExecStart=/usr/bin/dockerd --insecure-registry=47.96.71.197:90" 中的 --insecure-registry的值为harbor地址。并执行 1、sudo systemctl daemon-reload 2、sudo systemctl restart docker

镜像列表展示当前可用来创建蜜罐的镜像列表

!!! 注意：这里可以对镜像的端口进行设置、当前端口是确定的、除非你明确修改的意义、否则请不要随意修改

系统设置 / 镜像列表

镜像名称	镜像地址	镜像端口	镜像类型	系统类型	操作
ehoney/tomcat	47.96.71.197:90/ehoney/tomcat:v1	8080	http	Centos	<a href="#">编辑</a>
ehoney/telnet	47.96.71.197:90/ehoney/telnet:v1	23	telnet	Centos	<a href="#">编辑</a>
ehoney/ssh	47.96.71.197:90/ehoney/ssh:v1	22	ssh	Centos	<a href="#">编辑</a>
ehoney/redis	47.96.71.197:90/ehoney/redis:v1	6379	redis	Centos	<a href="#">编辑</a>
ehoney/mysql	47.96.71.197:90/ehoney/mysql:v1	3306	mysql	Centos	<a href="#">编辑</a>

### 进行协议转发的服务

系统设置 / 协议配置

服务类型	模块目录
http	/home/ehoney_proxy/httpproxy
telnet	/home/ehoney_proxy/telnetproxy
redis	/home/ehoney_proxy/redisproxy
ssh	/home/ehoney_proxy/sshproxy
mysql	/home/ehoney_proxy/mysqlproxy

### 密签跟踪服务器设置

The screenshot shows the CHONEY web-based management interface. The left sidebar contains a navigation menu with the following items:

- 密签管理
- 影子代理
- 诱捕管理
- 告警管理
- 系统设置 (System Settings)
- 镜像源配置
- 镜像列表
- 协议配置
- 密签配置 (Signature Configuration) - This item is highlighted with a blue background.

The main content area has a header "系统设置 / 密签追踪url配置". Below it is a form field labeled "密签追踪url" containing the value "http://192.168.22.176:5000". A blue "更新" (Update) button is located below the input field.

最后更新: 2021年7月9日

## 5. 模拟攻击

### 5.1 HTTP

#### 5.1.1 HTTP

通过终端或其他方式尝试对建立了透明转发到协议转发（HTTP）链路的探针进行http探测

```
(base) ~ curl http://192.168.22.246:1084

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <title>Apache Tomcat/8.0.23</title>
    <link href="favicon.ico" rel="icon" type="image/x-icon" />
    <link href="favicon.ico" rel="shortcut icon" type="image/x-icon" />
    <link href="tomcat.css" rel="stylesheet" type="text/css" />
  </head>

  <body>
    <div id="wrapper">
      <div id="navigation" class="curved container">
        <span id="nav-home"><a href="http://tomcat.apache.org/">Home</a></span>
```

在攻击列表中可查看具体的攻击详情信息、详情可查看http连接具体的信息

告警管理 / 告警列表

攻击IP	攻击跳转IP	蜜罐IP	被攻击服务	攻击IP地理位置	攻击时间	操作
192.168.22.246	192.168.22.176	10.42.0.60	http	局域网-局域网	2021-06-29 14:15	<a href="#">详情</a>
192.168.7.232	192.168.22.246	10.42.0.60	http	局域网-局域网	2021-06-29 14:15	<a href="#">详情</a>
192.168.22.246	192.168.22.176	10.42.0.57	ssh	局域网-局域网	2021-06-29 13:11	<a href="#">详情</a>
192.168.22.246	192.168.22.176	10.42.0.57	ssh	局域网-局域网	2021-06-29 13:09	<a href="#">详情</a>
192.168.22.246	192.168.22.176	10.42.0.57	ssh	局域网-局域网	2021-06-29 13:09	<a href="#">详情</a>
192.168.22.246	192.168.22.176	10.42.0.57	ssh	局域网-局域网	2021-06-29 13:09	<a href="#">详情</a>
192.168.22.246	192.168.22.176	10.42.0.57	ssh	局域网-局域网	2021-06-29 13:09	<a href="#">详情</a>
192.168.22.246	192.168.22.176	10.42.0.57	ssh	局域网-局域网	2021-06-29 13:09	<a href="#">详情</a>
192.168.22.246	192.168.22.176	10.42.0.57	ssh	局域网-局域网	2021-06-29 13:09	<a href="#">详情</a>
192.168.22.246	192.168.22.176	10.42.0.57	ssh	局域网-局域网	2021-06-29 13:09	<a href="#">详情</a>

共 31 条 < 1 2 3 4 > 前往  页

最后更新: 2021年6月30日

## 5.2 SSH

### 5.2.1 SSH

在终端尝试对建立了透明转发到协议转发（SSH）链路的探针进行ssh连接

```
(base) ~ ssh root@192.168.22.246 -p 1063
root@192.168.22.246's password:
Welcome to Alpine!

The Alpine Wiki contains a large amount of how-to guides and general
information about administrating Alpine systems.
See <http://wiki.alpinelinux.org>.

You can setup the system with the command: setup-alpine

You may change this message by editing /etc/motd.
```

在攻击列表中可查看具体的攻击详情信息、详情可查看ssh连接的账户密码

攻击IP	攻击跳转IP	蜜罐IP	被攻击服务	攻击IP地理位置	攻击时间	操作
192.168.22.246	192.168.22.176	10.42.0.57	ssh	局域网-局域网	2021-06-29 13:11	详情
192.168.22.246	192.168.22.176	10.42.0.57	ssh	局域网-局域网	2021-06-29 13:09	详情
192.168.22.246	192.168.22.176	10.42.0.57	ssh	局域网-局域网	2021-06-29 13:09	详情
192.168.22.246	192.168.22.176	10.42.0.57	ssh	局域网-局域网	2021-06-29 13:09	详情
192.168.7.232	192.168.22.246	10.42.0.57	ssh	局域网-局域网	2021-06-29 13:09	详情
192.168.22.246	192.168.22.176	10.42.0.57	ssh	局域网-局域网	2021-06-29 13:09	详情
192.168.22.246	192.168.22.176	10.42.0.57	ssh	局域网-局域网	2021-06-29 13:09	详情
192.168.22.246	192.168.22.176	10.42.0.57	ssh	局域网-局域网	2021-06-29 13:09	详情
192.168.22.246	192.168.22.176	10.42.0.57	ssh	局域网-局域网	2021-06-29 13:09	详情

最后更新: 2021年6月30日

## 5.3 MySQL

### 5.3.1 MySQL

通过终端或其他方式尝试对建立了透明转发到协议转发（mysql）链路的探针进行mysql探测

```
mysql -u root -h 192.168.22.246 -P 1085 -p 123456 -e "show databases"
Enter password:
ERROR 1045(28000): Access Denied for user root@10.2.2.2 (using password
YES)
```

在攻击列表中可查看具体的攻击详情信息、详情可查看mysql流量的具体信息

攻击IP	攻击跳转IP	蜜罐IP	被攻击服务	攻击IP地理位置	攻击时间	操作
192.168.7.232	192.168.22.246	10.42.0.61	mysql	局域网-局域网	2021-06-29 14:25	<a href="#">详情</a>
192.168.22.246	192.168.22.176	10.42.0.61	mysql	局域网-局域网	2021-06-29 14:25	<a href="#">详情</a>
192.168.7.232	192.168.22.246	10.42.0.61	mysql	局域网-局域网	2021-06-29 14:25	<a href="#">详情</a>
192.168.22.246	192.168.22.176	10.42.0.61	mysql	局域网-局域网	2021-06-29 14:25	<a href="#">详情</a>
192.168.22.246	192.168.22.176	10.42.0.61	mysql	局域网-局域网	2021-06-29 14:24	<a href="#">详情</a>
192.168.7.232	192.168.22.246	10.42.0.61	mysql	局域网-局域网	2021-06-29 14:24	<a href="#">详情</a>
192.168.22.246	192.168.22.176	10.42.0.61	mysql	局域网-局域网	2021-06-29 14:24	<a href="#">详情</a>
192.168.7.232	192.168.22.246	10.42.0.61	mysql	局域网-局域网	2021-06-29 14:24	<a href="#">详情</a>
192.168.22.246	192.168.22.176	10.42.0.61	mysql	局域网-局域网	2021-06-29 14:24	<a href="#">详情</a>
192.168.7.232	192.168.22.246	10.42.0.61	mysql	局域网-局域网	2021-06-29 14:24	<a href="#">详情</a>

最后更新: 2021年6月30日

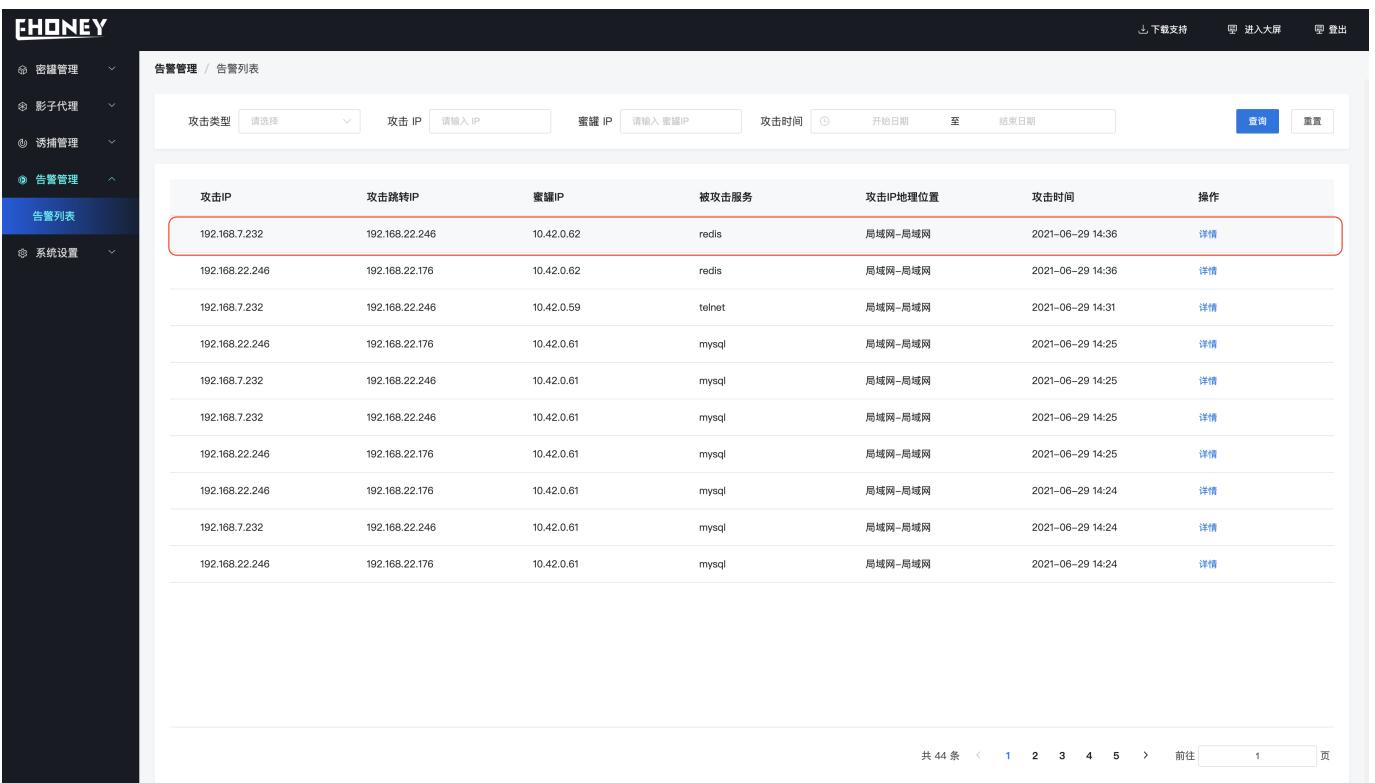
## 5.4 REDIS

### 5.4.1 REDIS

通过终端或其他方式尝试对建立了透明转发到协议转发（redis）链路的探针进行redis探测

```
redis-cli -h 192.168.22.246 -p 1087
192.168.22.246:1087> PING
PONG
```

在攻击列表中可查看具体的攻击详情信息、详情可查看redis流量的具体信息



The screenshot shows the CHONEY network monitoring interface. On the left is a sidebar with navigation links: 密鑑管理, 影子代理, 诱捕管理, 告警管理 (highlighted in blue), and 系统设置. The main area has a title '告警管理 / 告警列表'. Below it is a search bar with fields for '攻击类型' (Attack Type), '攻击 IP' (Attack IP), '蜜罐 IP' (Honeypot IP), '开始日期' (Start Date), '结束日期' (End Date), and buttons for '查询' (Search) and '重置' (Reset). A table lists 14 rows of attack logs. The first row is highlighted with a red border. The columns are: 攻击IP, 攻击跳转IP, 蜜罐IP, 被攻击服务, 攻击IP地理位置, 攻击时间, and 操作 (Details). The first row's details are: 192.168.7.232, 192.168.22.246, 10.42.0.62, redis, 局域网-局域网, 2021-06-29 14:36, and a '详情' (Details) link.

攻击IP	攻击跳转IP	蜜罐IP	被攻击服务	攻击IP地理位置	攻击时间	操作
192.168.7.232	192.168.22.246	10.42.0.62	redis	局域网-局域网	2021-06-29 14:36	<a href="#">详情</a>
192.168.22.246	192.168.22.176	10.42.0.62	redis	局域网-局域网	2021-06-29 14:36	<a href="#">详情</a>
192.168.7.232	192.168.22.246	10.42.0.59	telnet	局域网-局域网	2021-06-29 14:31	<a href="#">详情</a>
192.168.22.246	192.168.22.176	10.42.0.61	mysql	局域网-局域网	2021-06-29 14:25	<a href="#">详情</a>
192.168.7.232	192.168.22.246	10.42.0.61	mysql	局域网-局域网	2021-06-29 14:25	<a href="#">详情</a>
192.168.7.232	192.168.22.246	10.42.0.61	mysql	局域网-局域网	2021-06-29 14:25	<a href="#">详情</a>
192.168.22.246	192.168.22.176	10.42.0.61	mysql	局域网-局域网	2021-06-29 14:25	<a href="#">详情</a>
192.168.22.246	192.168.22.176	10.42.0.61	mysql	局域网-局域网	2021-06-29 14:24	<a href="#">详情</a>
192.168.7.232	192.168.22.246	10.42.0.61	mysql	局域网-局域网	2021-06-29 14:24	<a href="#">详情</a>
192.168.22.246	192.168.22.176	10.42.0.61	mysql	局域网-局域网	2021-06-29 14:24	<a href="#">详情</a>
192.168.7.232	192.168.22.246	10.42.0.61	mysql	局域网-局域网	2021-06-29 14:24	<a href="#">详情</a>
192.168.22.246	192.168.22.176	10.42.0.61	mysql	局域网-局域网	2021-06-29 14:24	<a href="#">详情</a>
192.168.7.232	192.168.22.246	10.42.0.61	mysql	局域网-局域网	2021-06-29 14:24	<a href="#">详情</a>
192.168.22.246	192.168.22.176	10.42.0.61	mysql	局域网-局域网	2021-06-29 14:24	<a href="#">详情</a>

共 44 条 < 1 2 3 4 5 > 前往 1 页

最后更新: 2021年6月30日

## 5.5 TELNET

### 5.5.1 TELNET

通过终端或其他方式尝试对建立了透明转发到协议转发（telnet）链路的探针进行telnet探测

```
(base) ~ telnet 192.168.22.246 1086
Trying 192.168.22.246...
Connected to 192.168.22.246.
Escape character is '^]'.
Ubuntu 17.10
telnet-7b8bbf4958-tph52 login:
```

在攻击列表中可查看具体的攻击详情信息、详情可查看telnet流量的具体信息

攻击IP	攻击跳转IP	蜜罐IP	被攻击服务	攻击IP地理位置	攻击时间	操作
192.168.7.232	192.168.22.246	10.42.0.59	telnet	局域网-局域网	2021-06-29 14:31	<a href="#">详情</a>
192.168.22.246	192.168.22.176	10.42.0.61	mysql	局域网-局域网	2021-06-29 14:25	<a href="#">详情</a>
192.168.7.232	192.168.22.246	10.42.0.61	mysql	局域网-局域网	2021-06-29 14:25	<a href="#">详情</a>
192.168.7.232	192.168.22.246	10.42.0.61	mysql	局域网-局域网	2021-06-29 14:25	<a href="#">详情</a>
192.168.22.246	192.168.22.176	10.42.0.61	mysql	局域网-局域网	2021-06-29 14:25	<a href="#">详情</a>
192.168.22.246	192.168.22.176	10.42.0.61	mysql	局域网-局域网	2021-06-29 14:24	<a href="#">详情</a>
192.168.7.232	192.168.22.246	10.42.0.61	mysql	局域网-局域网	2021-06-29 14:24	<a href="#">详情</a>
192.168.22.246	192.168.22.176	10.42.0.61	mysql	局域网-局域网	2021-06-29 14:24	<a href="#">详情</a>
192.168.7.232	192.168.22.246	10.42.0.61	mysql	局域网-局域网	2021-06-29 14:24	<a href="#">详情</a>
192.168.7.232	192.168.22.246	10.42.0.61	mysql	局域网-局域网	2021-06-29 14:24	<a href="#">详情</a>

最后更新: 2021年6月30日

## 6. 其他

---

### 6.1 名字解释

---

#### 蜜罐

- 容器化部署、全平台高交互蜜罐
- 仿真业务、协议等、吸引黑客攻击
- 对攻击行为进行捕获和分析、了解攻击方所使用的工具与方法
- 推测攻击意图和动机

#### 蜜网

- 是一种新型的蜜罐系统架构、通过将蜜罐集中的部署在一个受控的蜜网中、统一的进行数据分析和产生告警、而在真实的生产网络区域中仅仅部署一些轻量级的重定向模块、探针会通过监听相应的端口、将自己伪装成一个蜜罐，当探针受到攻击后会透明的将攻击流量转发到蜜场的蜜罐中。

#### 探针

- 应用服务器、是黑客的第一道攻击入口

#### 密签

- 定位攻击者 数据、文件、程序里加入特定标记，识别数据泄露，支持文件密签、邮件密签、图片密签、DNS密签、git密签、sql密签等。比如，一个Word文档或是一个Windows文件夹对于大部分人来说是无害的，安全的，我们利用了它们的一些可访问网络资源的特性，改造成密签，当有黑客从被攻击的服务器或蜜罐中下载后打开文档或者是进入文件夹的时候就会触发告警。

#### 诱饵

- 诱惑攻击者 用于迷惑攻击者的数据包括文件、数据库、配置、日志、代码等信息，诱使攻击者对密罐进行攻击。

#### Agent

- 攻击流量透明转发 agent通过绑定转发仿真服务蜜罐，将TCP、UDP、ICMP、SYN等类型的攻击流量透明、无感知的被转发到蜜网中的蜜罐里，此时的蜜罐可以捕获攻击者多种操作行为，同时发现新的攻击方式和漏洞利用方法
- 诱饵、密签下发 agent接收下发策略，把策略管控端的诱饵、密签下发到服务器，诱捕引导黑客攻击到蜜网的蜜罐

#### 协议转发

- 协议代理转发 agent通过绑定转发仿真服务蜜罐，攻击者的攻击流量将会通过对称的网络协议（比如HTTP、SSH等）被转发到蜜网中的蜜罐里，此时的转发代理可以实时获取攻击命令、目录爆破探测、登录爆破、ssh异常连接等攻击行为

#### 透明转发

- 探针服务器到蜜网服务器的端口转发

最后更新: 2021年6月30日

## 6.2 FAQ

---

- 部署服务器对配置有什么要求？
1. 系统要求CentOS 7以上，内存4G、磁盘空间10G以上
  2. 3306、6379、5000、8080、8082端口未被使用
- 一键安装成功，但是浏览器无法访问？
1. 检查docker容器状态是否正常 `docker ps`
  2. 检查web服务器容器日志 `docker logs -f $(docker ps | grep decept-defense:latest | awk '{print $1}')`
- 为什么模拟攻击没有攻击日志？
1. 通过透明代理列表，网络探测检查攻击IP、端口可以访问。
  2. 确保蜜网服务器上/home/ehoney\_proxy目录下对应的代理文件存在。
- 为什么我部署了多台探针Agent，但是探针列表只显示一个？
1. 删除agent/conf目录下agent文件，重新启动agent进程，确保每台服务器上agentID不一致。
- 为什么蜜罐列表中新建蜜罐失败？
1. 检查蜜网服务器资源状态，有可能是磁盘、cpu利用率满了。一般一台2核4G服务器上最多支持部署20个蜜罐。
  2. 如果访问harbor不是https协议，需要在docker启动的时候指定insecure-registry。可以通过 `kubectl describe pod` 查看具体报错信息。
- 为什么蜜罐拓扑图不显示蜜罐和相关连线？
1. 蜜罐拓扑图是以协议代理、透明代理为维度，如果没有代理则不会显示相关蜜罐。
- 为什么我部署了探针但是探针列表却不显示？
1. 检查agent中配置是否正确。主要检查conf/agent.json中strategyAddr(redis地址)、strategyPass(redis密码)、sshKeyUploadUrl(服务端更新sshkey地址)。
  2. 确认探针服务器和web/蜜网服务器网络可以通信。
- 为什么协议代理创建失败？
1. 查看协议代理日志/home/relay/agent/proxy/log/proxy.log
  2. 检查转发端口是否被占用/ssh\_proxy文件是否正确
- 配置修改
1. redis、mysql修改请在部署服务前、修改源代码conf目录下的app.conf分别修改mysql或redis条目
  2. 同时、当前版本暂时不支持对后端web服务占用端口的修改、后续版本会支持
- 

最后更新: 2021年7月7日



<https://github.com/seccome/Ehoney/>